

## Claim Amendments

Please amend the claims to be as follows.

1. (previously presented) A method of compiling a computer program with inline specialization, the method comprising:

given a call-graph, if multiple call-chains in the call-graph have a common call site, inlining the common call site in one or more of the call-chains, without inlining the common call site into all of said multiple call-chains having the common call site.

2. (currently amended) The method of claim 1, further comprising:  
whenever a call site from routine x to routine y is inlined, new [[edges]] call sites are added from routine x to all routines inlinable within routine y.
3. (currently amended) The method of claim 2, further comprising:  
materialization of summary information for the new call sites added to the call-graph.
4. (currently amended) The method of claim 3, further comprising:  
addition of the new call sites to [[the]] a global work-list so that [[these]] the new call sites are considered for inlining.
5. (currently amended) The method claim 4, further comprising:  
addition of dependence relationships between call sites~~[[. If]]~~, wherein if a new call site, y, is added because of inlining of call site, x, then y is dependent on x.
6. (currently amended) The method of claim 5, further comprising:  
patching of the new call site, y, during inline transformation of call site, x, ~~with the aim of generating the~~ and generating an intermediate transformation for the new call site, y.

7. (currently amended) An apparatus for compiling a computer program with inline specialization, the apparatus comprising:
  - memory configured to store computer-readable instructions and data;
  - a processor configured to access said memory and to execute said computer-readable instructions;
  - [[means]] computer-readable instructions stored in said memory and configured to inline a common call site in one or more call-chains in a call-graph, without inlining the common call site into all call-chains having the common call site.
8. (currently amended) The apparatus of claim 7, wherein whenever a call site from routine x to routine y is inlined, new [[edges]] call sites are added from routine x to all routines inlinable within routine y.
9. (currently amended) The apparatus of claim 8, wherein materialization of summary information for the new call sites added to the call-graph is performed.
10. (currently amended) The apparatus of claim 9, wherein the new call sites are added to [[the]] a global work-list so that [[these]] the new call sites are considered for inlining.
11. (original) The apparatus of claim 10, wherein dependence relationships are created between call sites.
12. (currently amended) The apparatus of claim 11, wherein the inline transformation patches up [[the]] an intermediate representation of the new call sites (by considering the dependence relationships) before potentially inlining [[them]] the new call sites.

13. (previously presented) A computer program product comprising a computer-usable medium having computer-readable code embodied therein, the computer program product being a source code compiler with cross-module optimization, the compiler including an inline specialization feature such that given a call-graph, if multiple call-chains in the call-graph have a common call site, the common call site is inlined in one or more of the call-chains, without being inlined into all of the multiple call-chains having the common call site.